PicoLAS

FOCUSSING POWER TO THE POINT

User Manual

Rev. 09.01 valid from June 2009

PLCS-21

PicoLAS GmbH
Company for Innovative Power Electronics und Laser Technology

Kaiserstraße 100
52134 Herzogenrath

Tel: +49(0)2407-563 58-0
Fax: +49(0)2407-563 58-29
E-Mail: info@picolas.de
Web: www.picolas.de

PicoLAS
FOCUSSING POWER TO THE POINT

Table of Contents

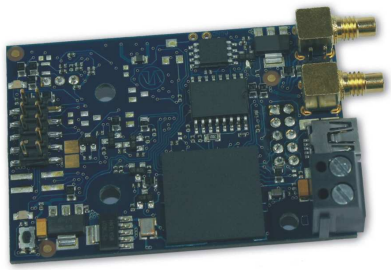
PLCS-21 3
Description of Connections 4
How to get started 5
Operating Modes 6
Using the PLCS-21 as a Control Unit for a Laser Diode Driver 6
Using the PLCS-21 as a Digital Function Generator 7
Calibration 7
Trigger Modes 8
Controlling the PLCS-21 Using a PLB-21 10
Controlling the PLCS-21 via USB 13
Electrical Characteristics 31
Absolute Maximum Ratings 31

PicoLAS
FOCUSSING POWER TO THE POINT

Rev. 09.01 valid from April 2009

PLCS-21

Providing Brains for pulsed Driver Modules



- Control of pulse current, pulse width and repetition rate
- Microcontroller supervision of all pulser activity
- Provides USB interface for communication with PC and interface for PLB-21
- Several additional failsafe mechanisms to protect your laserdiodes
- Standalone Pulse-Generator capability

Product Description:

Many PicoLAS driver modules for pulsed lasers can be upgraded with the PLCS-21 optional piggyback-controller. This versatile funtion generator allows full control of pulse width, pulse current and repetition rate.
Priority is given to provide maximum safety for the expensive laser diodes. An on-board micro-controller monitors current, voltage, temperature, pulse duration and rep-rates and cuts power to the diode if necessary.
The PLCS-21 provides a galvanically isolated USB interface. With this, the upgraded driver module can be connected to a PC. An additional interface can be used to connect the external operating unit PLB-21. This allows easy monitoring and manipulating the driver's behavior.
The piggyback controller also provides several internal trigger mechanism. This eliminates the need for an external function generator as a trigger device.
The PLCS-21 can also work as a standalone digital function generator. A square-wave signal with pulse width from 2 ns up to one second can be drawn from a coaxial 50 Ohm output.
The PLCS-21 automatically identifies the connected driver module. Maximum pulsewidth, output power
etc. depend on the used laser diode driver.**

Technical Data:*

Supply voltage	+15 V (supply voltage for LDP-V is fed through PLCS)
Min. Pulse width	2 ns
Max. Pulse width	1 s
Pulse width adjustment	in steps of 1ns (<250ns) in steps of 5ns (>250ns)
Min. Repetition rate	1 Hz
Max. Repetition rate	2.4 Mhz
Repetition Rate adj.	in steps of 1 Hz
Coaxial 50 Ohm output	Generator Voltage: 10V Maximum Load: 50 Ohm
Trigger Inputs	50 Ohm, 5V, SMC connector 500 Ohm, 5V, 2-Pin connector
Interfaces	USB 2.0, PLB-20
Dimensions	67,7 x 42,3 x 22 mm
Weight	26 g
Operating temperature	0 to +55 °C

* Technical data is subject to change without further notice.
** See manuals for details.

Optional Accessories: PLB-21
Compatible Products: LDP-V 03-100 V3
LDP-V 50-100 V3
LDP-V 240-100 V3
LDP-AV D06-N20

Description of Connections

The following drawing shows all connections which are available to the user.

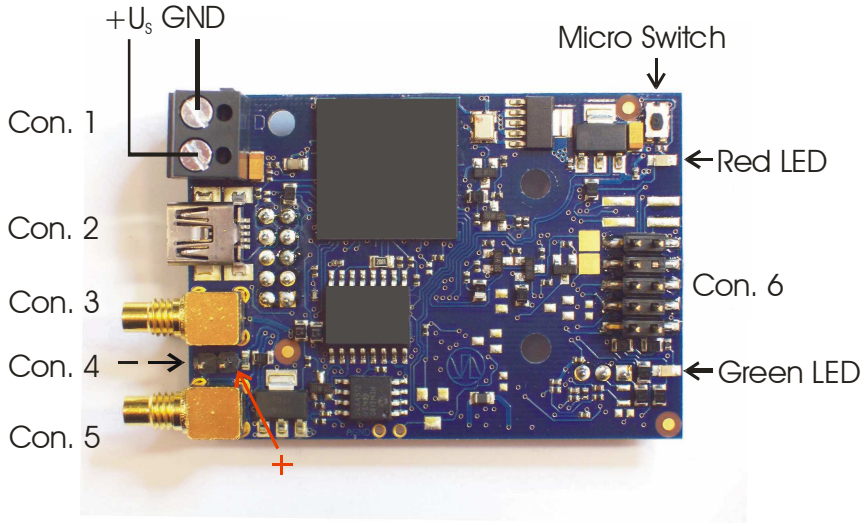


Figure 1: Connectors of the PLCS-21

Connector 1	Supply Voltage
Connector 2	Mini USB Connector
Connector 3	SMC Connector, 50 Ohm pulse output
Connector 4	2-Pin Connector, 500 Ohm trigger input
Connector 5	SMC Connector, 50 Ohm trigger input
Connector 6	Connector for PLB-20

Micro-Switch: no function, reserved for future use.

Green LED:
- On: OK
- Off: PLCS-21 not operational

Red LED:
- On: Error
- Blinking: Calibration in progress

The connectors on the bottom side of the device allow connecting the PLCS-21 to a PicoLAS laser diode driver, they must not be used for any other purpose.

How to get started

Step #	What to do	Note
1	Unpack your Device	
2	Optional: Connect your Scope to the current monitor of your diode driver, as the mounted PLCS-21 covers the SMC jacket	
3	Mount the PLCS-21 to your PicoLAS laser diode driver. (e.g. LDP-V 50-100 V3)	
4	Connect the power supply	See *Figure 1* for details
5	Connect the PLCS-21 either to the PLB-21 or a PC via USB	See *Figure 1* for details
6	Optional: Connect an external trigger source	See *Figure 1* for details
7	Power on your device	
9	When the initializing is done, adjust the pulse parameters to your needs. See "Controlling the PLCS-21 using the PLB-21 / USB" and the device specific manual for details.	
9	Activate the output	

Operating Modes

The PLCS-21 can be operated in two different ways: As a controller for a compatible PicoLAS diode driver and as a stand-alone digital function generator. If used as a controller it supports the **Voltagemode** and the **Currentmode**, otherwise the **Frequency Generator** mode. In either mode a number of trigger modes and thee different control interfaces are supported. See chapter 6 for a detailed description of the trigger modes and chapters 7 and 8 for the control interfaces.

Using the PLCS-21 as a Control Unit for a Laser Diode Driver

First the PLCS-21 must be mounted on the laser diode driver by using the two connectors on its bottom side and three screws M3 x 8mm. PLCS-21 and the diode driver are both powered by a single 15 V power supply via connector 1. Then a PLB-21 or a PC with appropriate software must be connected to the PLCS-21. Now the PLCS-21 is ready for use.

> Important: Connector 3 must never be connected to any cable if the PLCS-21 is mounted on a diode driver. Anything else can result in deformed laser pulses. Though monitored by the PLCS-21 the power being fed into a laser diode may exceed the adjusted value and can, in some cases, destroy your diode. Correct operation is guaranteed only if connector 3 is left unconnected.
> Furthermore the connected diode driver must use its internal high voltage source. See device specific datasheet for instructions how to enable the internal HV source.

Voltage Mode

This mode is active by default once a pulser is connected to the PLCS-21. The pulse width, repetition rate, voltage, temperature monitor and over-current detection of the connected pulser can be set. In Voltage Mode it is also possible to carry out a calibration. See chapter "Calibration" for details. When it is done, a change to Current Mode can be carried out.

Current Mode

When calibration is successfully completed, a change to Current Mode can be carried out. Now, in addition to voltage, the current can alternatively be set. These two parameters are linked together in such a way that a change to the set current always generates a change to the voltage. The minimum value for the voltage is specified by U_{min}; the maximum value for the current is specified by I_{max}. The interdependency of both values is determined during calibration and is stored in the PLCS-21.
If another diode is connected to the pulser or another pulser is connected to the PLCS-21 then the calibration must be manually repeated. A change in the diode or the pulser (with the same type) cannot be detected by the PLCS-21. If the values of U_{min} or I_{max} change, then this also means that a new calibration must be carried out in order to take the new limits into account.

Using the PLCS-21 as a Digital Function Generator

The PLCS-21 will automatically work in this mode if it is not connected to a laser diode driver. Only the functions for setting the pulse width, repeat rate, trigger modes and activating/ deactivating of the output are active. All others are not used and the corresponding LSTAT and ERROR bits can be ignored. The generated signal can be received from connector 3. The schematic of output circuit is shown in *Figure 2*. The output amplifier will generate a square-wave signal with an amplitude of 10V. If a 50 Ohm load is attached to connector 3, this will result in a signal level of 5V at the load. Unlike the trigger inputs the output circuit is not galvanically isolated from the power supply. To obtain a well-formed signal a load of 50 Ohm is recommended. Refer to the electrical characteristics on chapter "Electrical Characteristics" for further details.

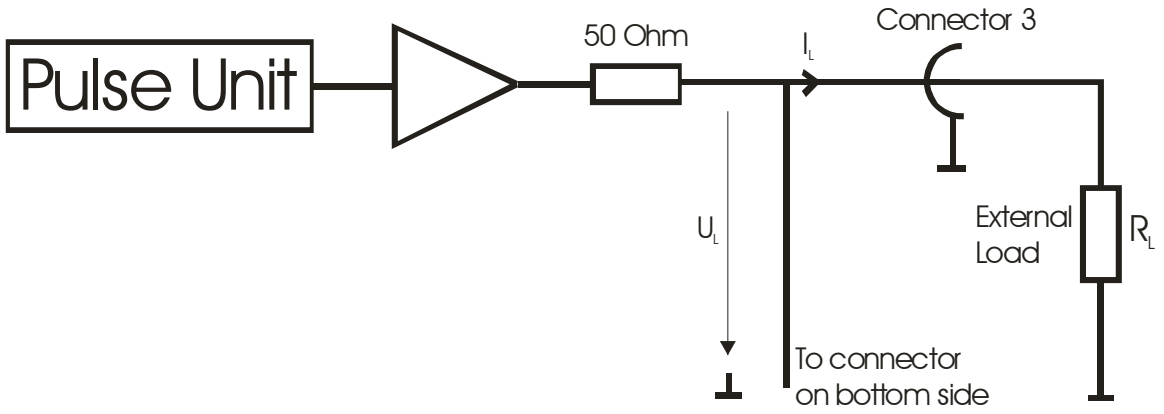


Figure 2: Pulse output circuit

Calibration

The PLCS-21 can calibrate itself to the characteristics of the connected laser diode. If the calibration is successfully performed the user is able to switch into the current mode in which either the precharge voltage or the diode current can be adjusted.
In order to perform a calibration a valid maximum diode current must be set. This value will be the maximum current for the given diode. Note that a maximum of 99% of this value will be reached during calibration in order to protect the diode. Furthermore a valid minimum voltage (U_{min}) must be supplied. Most times this will be slightly above the threshold voltage of the connected diode. The user must assure that with this voltage the minimum load current of the connected diode driver is reached. See the device specific datasheet for the actual minimum current. If the minimum load current is not reached during the start of calibration, an error message is displayed and the user needs to increase U_{min}.
During the calibration the PLCS-21 generates a linear list of measuring points from which it analyses the relation of a given voltage to the measured current. If successfully performed it saves the collected data, which is used to calculate a precharge voltage for a given current. Now the user can not only set a precharge voltage, but also a set point current.
Note that a calibration is only valid for a given hardware setting. The PLCS-21 cannot detect any change in the connected laser diode. So the user must recalibrate it every time the connected hardware is changed.

Trigger Modes

The PLCS-21 supports a number of trigger modes which are described below. The width and repetition rate of the pulses generated are user defined. Pulses will always be generated as long as the trigger condition matches and the laser is enabled.
As an input for the trigger signal the connector 4 or 5 can be used. *Figure 3* shows the schematic of both inputs. Note that they are galvanically isolated from the supply voltage. For trigger levels see the electrical characteristics on chapter "Electrical Characteristics".

> Important: Never use both trigger inputs at the same time. Correct operation is not ensured if both inputs are connected to a source. Furthermore, a signal fed into one input may result in a current flowing out of the other input. This might damage your trigger source.

In the following the different trigger modes are described separately:

Edge

In this mode an external trigger source is required to generate pulses. The pulses can either be generated on the rising or the falling edge of the supplied trigger. On each edge which equates the given setting, a given number of pulses ("Shots") will be generated.

Pulse

In this mode an external trigger source is required to generate pulses. The PLCS will generate pulses during the positive or negative part of the trigger source.

Internal

In this mode the external trigger source is ignored. The PLCS will generate an infinite number of pulses by itself.

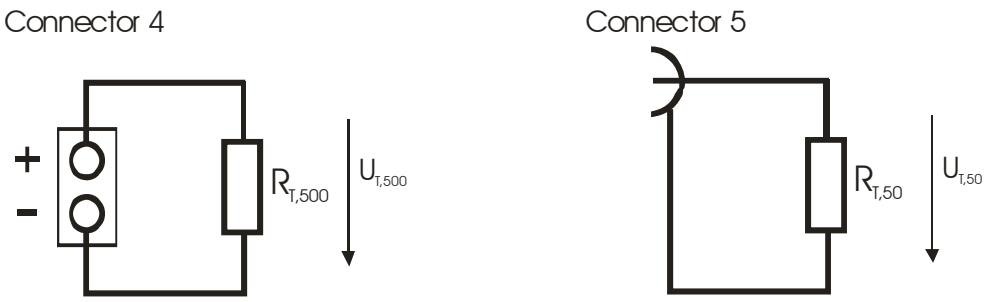


Figure 3: Trigger input circuit

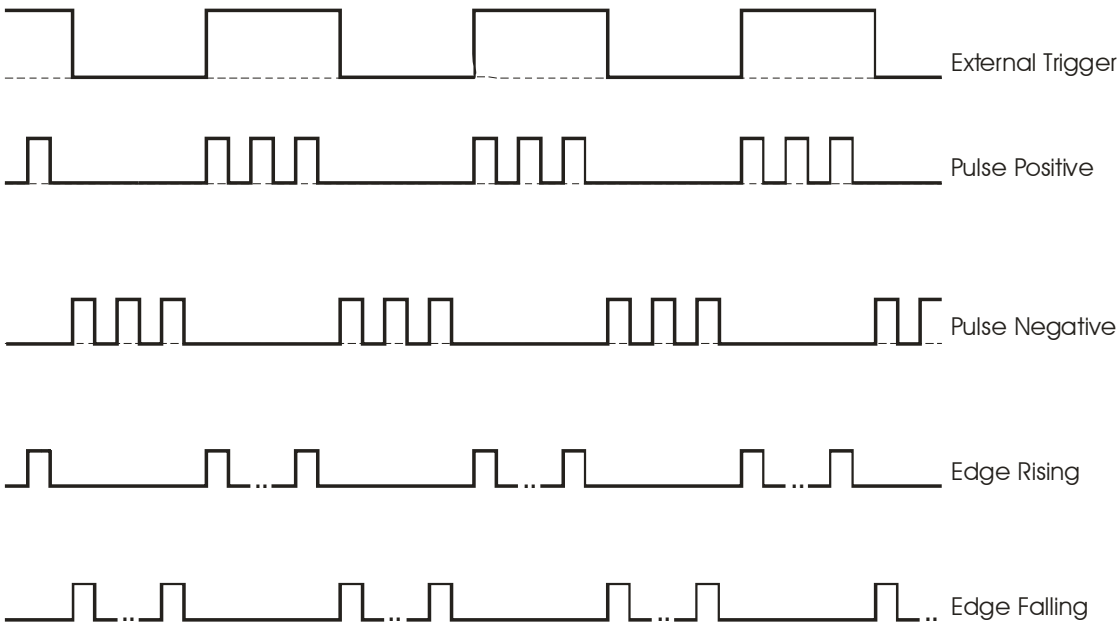


Figure 4: Shematic voltage diagram

PicoLAS
FOCUSSING POWER TO THE POINT

Controlling the PLCS-21 Using a PLB-21

To control the PLCS-21 with a PLB-21 it must be connected via the enclosed cable. The PLB-21 will not work if both, the USB and the PLB-21, are connected the same time.
When the PLB-21 is connected the first time to a PLCS-21 you are asked to download a new driver. This must be confirmed with "yes" for the PLB-21 to work properly.

Menu Structure

The following diagram shows the structure of the PLB-21 menu which affects the PLCS-21. All entrys are described in detail. All other menu entries are described in the PLB-21 manual. For detailed instructions see the PLB-21 manual.

Menu root

- Pulseparameter
  - Width
  - Reprate
  - Current
  - Voltage
- PLCS Config
  - Mode
  - Imax
  - Umin
  - Calibrate
- Trigger
  - Mode
  - Logic
  - Shots
- Temperature
  - Dev. Off
  - Dev. Max
  - Dev. Act
  - PLCS Act.
- LDP Info/Config
  - Name
  - Reset to def.

Pulseparameter

In this menu point you can modify the pulse length, repetition rate, current and voltage of the pulse. The minimum and maximum values for every entry depend on your hardware configuration. The PLCS-21 automatically detects the connected laser diode driver and sets the values according to it. See the device specific datasheet for detailed information.

Width
This value defines the pulse width in nano seconds (ns). The minimum and maximum values are defined by the connected pulser and by the actual repetition rate. In order to generate the absolute maximum width, the user may need to reduce the repetition rate.

Reprate
This value defines the repetition rate in Hertz (Hz). The actual minimum and maximum values depend on the connected pulser an on the given pulse width. In order to generate the absolute maximum repetition rate the user may need to reduce the pulse width.

Current
The current is only used in current mode. After a successfully performed calibration, the user can set up a current which flows through the connected diode during the pulse. Note that the voltage changes according to the given current as the precharge voltage is automatically modified to match the set point current.

Voltage
This value defines the precharge voltage of the connected pulser. The actual current depends on the connected pulser and diode. The maximum voltage depends on the connected pulser. If the set point voltage generates a current flow greater than the maximum current defined by Imax, the laser will be disabled and an error message is displayed.

PLCS Config
In this submenu you can change the operation mode of the PLCS-21, the maximum pulse current and the minimum voltage for calibration. Furthermore you can execute a calibration.

Mode
The available operation modes depend on your hardware configuration. If you use the PLCS-21 as a stand alone device (without a connected laser diode driver) it can only be used as a frequency generator (“Freq. gen”), otherwise “Voltage” and/or “Current” are possible. Before you can select “Current” you must perform a calibration. See chapter “Operation modes” and “Calibration” for more information.

I_{max}
This value defines the maximum current output of the connected driver. If the given settings generate a higher current, the laser diode is automatically disabled. This requires no calibration. If a calibration is performed this value defines the maximum current for it.
When the PLCS-21 is used as a frequency generator I_{max} has no effect

U_{min}
This value defines the starting voltage for the calibration process. It must be above the threshold voltage of the connected laser diode. Otherwise the calibration will fail. In the modes “Voltage” or “Freq. gen” this setting has no effect.

Calibration
When selected a press on the “Enter” key will perform a calibration. See chapter “Calibration” for a detailed description.

Trigger

The PLCS-21 supports a number of trigger modes. For a detailed description of each mode see chapter “Trigger modes”.

Mode

This selects the used trigger mode. The modes are equal for all operation modes and connected drivers. Available are “edge”, “internal” and “pulse”.

Logic

This option is only used when the trigger mode is either “edge” or “pulse”. In “edge” mode you can select if pulses should be generated on the rising or falling edge of the supplied trigger. In “pulse” mode it selects weather pulses should be generated on “positive” (high) or the “negative” (low) part of the trigger signal.

Shots

When using the edge mode, the number of generated pulses can be determined by the user. The given number of pulses will always be generated, even if another trigger is received during generation.

Temperature

The PicoLAS laser diode drivers are equipped with a temperature sensor. In this submenu the minimum, maximum, current and shutdown temperatures can be read and modified.
All values are in °C.

Dev. Off

This shows the user defined shutdown temperature. If the diode driver reaches this temperature during operation, the laser will be disabled and an error message is displayed. It can be modified within the boundaries defined in the device specific datasheet.

Dev. Max

This is shows the maximum operating temperature of the connected diode driver. It also defines the maximum value for the shutdown tamerature.

Dev. Act

This displays the current temperature of the diode driver.

PLCS Act

This shows the current temperature of the PLCS-21. It should not exceed 80°C.

LDP Info/Config

This submenu holds some information and configuration options of the connected laser diode driver.

Name

This shows the name of the connected laser diode driver.

Reset to def.

This forces the PLCS-21 to reset all parameters to factory defaults. All settings and calibration data are lost.

Controlling the PLCS-21 via USB

Introduction

In addition to being able to connect up a PLB-21, the PLCS-21 also has a USB interface to communicate with a computer/laptop. This interface allows communications over both a serial text interface as well as using the PicoLAS protocol. While the text interface is designed for communication with a terminal program, the PicoLAS protocol is designed as a system interact protocol.

The switching between the two protocols occurs automatically as soon as the PLCS-21 receives a certain sequence. The corresponding commands are:

- PING for the PicoLAS protocol
- "init" followed by <Enter> for the text interface

If the PLB-21 and the USB interface are both connected at the same time then only the USB interface can be used. As soon as the USB connection is connected to a computer then the PLCS switches automatically over to this.

Description of the USB Interface

The USB connection of the PLCS-21 emulates a virtual COM port under Windows. The necessary drivers can be downloaded free of charge under http://www.ftdichip.com/Drivers/VCP.htm, or they are already included in the current Linux kernels. The virtual COM port created by this can be addressed like a regular one. The connection settings are:

Baud rate	115200
Data bits	8
Stop bits	1
Parity	even

PicoLAS
FOCUSSING POWER TO THE POINT

The Serial Text Interface

The following section describes the structure and commands of the text interface.

Structure

Every command that is sent to the PLCS-21 must be completed with a CR (Enter). It consists of a command word followed by a parameter. If the command was successfully executed then an "0" is sent, otherwise a "1". If the command requires an answer parameter, this parameter is sent before the confirmation is given.

Example:

The user would like to read out the voltage currently being used by the pulser.

User input: gvoltage<Enter>

Output of the PLCS-21: 12000

0

Input is done in ASCII code and is case sensitive. Every terminal can be used that supports this standard.

Commands for the PLCS-21

The following table contains a command reference for the PLCS-21.

Command	Parameter	Answer	Description
help	--	Help text	Output of a help text
spulse	Pulse length in ns	--	Sets the length of the pulse to be emitted
gpulse	--	Pulse length in ns	Outputs the current pulse length
gpulsemin	--	Pulse length in ns	Outputs the minimum pulse length
gpulsemax	--	Pulse length in ns	Outputs the maximum pulse length
sreprate	Repeat rate in Hz	--	Sets the repeat rate of the pulse
greprate	--	Repeat rate in Hz	Outputs the current repeat rate
grepratemin	--	Repeat rate in Hz	Outputs the minimum repeat rate
grepratemax	--	Repeat rate in Hz	Outputs the maximum repeat rate
svoltage	Voltage in mV	--	Sets the precharger voltage of the pulser
gvoltage	--	Voltage in mV	Outputs the current precharger voltage of the pulser

Command	Parameter	Answer	Description
gvoltagemin	--	Voltage in mV	Outputs the minimum precharger voltage of the pulser
gvoltagemax	--	Voltage in mV	Outputs the maximum precharger voltage of the pulser
scurrent	Current in mA	--	Sets the pulse current to the indicated value [Only in Current Mode (mode 2)]
gcurrent	--	Current in mA	Outputs the present pulse current [Only in Current Mode (mode 2)]
gcurrentmin	--	Current in mA	Outputs the minimum pulse current [Only in Current Mode (mode 2)]
gcurrentmax	--	Current in mA	Outputs the maximum pulse current [Only in Current Mode (mode 2)]
sshots	Number of pulses	--	Sets the number of pulses to be emitted to the indicated value [Only in trigger mode 0 and 1]
gshots	--	Number of pulses	Outputs the currently set number of pulses
laseron	--	--	Activates pulse output
laseroff	--	--	Deactivates pulse output
strgmode	Trigger mode	--	Sets the trigger mode to be used (refer to trigger modes)
gtrgmode	--	Trigger mode	Outputs the current trigger mode
slstat	32 bit number	--	Sets the LSTAT register to the value
glstat	--	32 bit number	Outputs the LSTAT register
gerror	--	32 bit number	Outputs the ERROR register
clrerror	--	--	Resets the ERROR register
sumin	Voltage in mV	--	Sets the starting value for calibration
gumin	--	Voltage in mV	Outputs the starting value for calibration
socur	Current in mA	--	Sets the maximum current for over-current detection
gocur	--	Current in mA	Outputs the present value for over-current switching

Command	Parameter	Answer	Description
stempoff	Temperature in °C	--	Changes the switch-off temperature to the passed value
gtempoff	--	Temperature in °C	Outputs the current switch-off temperature
gtempoffmin	--	Temperature in °C	Outputs the minimum switch-off temperature
gtempoffmax	--	Temperature in °C	Outputs the maximum switch-off temperature
smode	Mode	--	Sets the operating mode to the passed value 0 = Frequency Generator 1 = Voltage Mode 2 = Current Mode
gmode	--	0…2	Outputs the current operating mode
calibrate	--	--	Carries out a calibration
default	--	--	Resets all parameters to factory defaults

If an Error Occurs

If an error occurs during operation the pulse output is switched off and a message is sent to the terminal. Errors have to be acknowledged with "clrerror" otherwise switching on again of pulse output is not possible. Note that warnings are also displayed this way but these do not switch off pulse output. Hence it is not necessary to aknowledge warnings with "clrerror".
This message has this format:

err: <Error Register>

The parameter <Error Register> represents the content of the ERROR register in binary form.

The PicoLAS Protocol

The following section describes the structure and possible commands of the PicoLAS protocol.

Structure

Each transmission consists of 12 bytes – called a frame as follows – which must be sent consecutively. Otherwise the system times out and the transmission must start again from the beginning.

A frame has a fixed structure. The first two bytes describe the command, the following eight bytes the parameters, followed by one reserved byte and one checksum byte. The checksum is calculated out of the first 11 bytes which are linked by a bitwise XOR.

Thus a frame has the following structure:

Byte	Meaning
1	Bit 8-15 of the command
2	Bit 0-7 of the command
3	Bit 56-63 of the parameter
4	Bit 48-55 of the parameter
5	Bit 40-47 of the parameter
6	Bit 32-39 of the parameter
7	Bit 24-31 of the parameter
8	Bit 16-23 of the parameter
9	Bit 8-15 of the parameter
10	Bit 0-7 of the parameter
11	Reserved, always 0x00
12	Checksum

A properly received frame must be acknowledged by the recipient with an answer, which is also a frame. If the acknowledgement does not occur then the command has not been processed and the sending procedure should be repeated.

If the recipient recognizes the command as valid, but not the parameters, then it will answer with a ILGLPARAM (0xFF12) as command.

In the case that the recipient receives an invalid command it will answer with UNCOM (0xFF13).

If a faulty checksum is recognized then the answer is RXERROR (0xFF10). If this error occurs often then the connection should be checked.

Using the REPEAT (0xFF11) command the recipient can instruct the sender to send the most recent frame again.

PicoLAS
FOCUSSING POWER TO THE POINT

General Commands

The following list contains an overview of the general commands which are supported by every product from PicoLAS which makes use of this protocol. The explanation of the individual commands is given further below.

Command Name	Sent Frame		Answer Frame	
	Command	Parameter	Command	Parameter
PING	0xFE01	0	0xFF01	0
IDENT	0xFE02	0	0xFF02	ID
GETHARDVER	0xFE06	0	0xFF06	Version
GETSOFTVER	0xFE07	0	0xFF07	Version
GETSERIAL	0xFE08	0 … 255	0xFF08	Refer to description
GETIDSTRING	0xFE09	0 … 255	0xFF09	Refer to description
GETDEVICECHECKSUM	0xFE0A	0	0xFF0A	CRC16 checksum
RESET	0xFE0E	0	0xFF0B	0

PING

Is used to determine the presence of a connected recipient and to initialize the interface of the recipient for this protocol. Has no effect on the condition of the recipient. The command parameter is always 0, the answer parameter too.

IDENT

It is used to determine the device ID of an attached recipient. Has no effect on the condition of the recipient. The parameter is always 0. The answer contains the ID.

GETHARDVER

Instructs the recipient to send back the version number of the hardware being used. The parameter is always 0. The answer contains the hardware version of the recipient. The format of the answer is: 0x000000<major><minor><revision>. In other words, one byte for each of the three elements of the version number.
As example, version 1.2.3 has the parameter 0x000000010203.

GETSOFTVER

Instructs the recipient to send back the version number of the software being used. The parameter is always 0.
The answer contains the software version of the recipient. The format of the answer is: 0x000000<major><minor><revision>. In other words, one byte for each of the three elements of the version number.
As example, version 2.3.4 has the parameter 0x000000020304.

GETSERIAL

Instructs the recipient to send back its serial number. If 0 is sent as parameter, the answer contains the number of (ASCII) digits of the serial number; otherwise the respective position of the serial number is sent in ASCII format.

GETIDSTRING

Instructs the recipient to send back the name of the device. If 0 is sent as parameter, the answer contains the number of digits of the string, otherwise the respective position of the serial number is sent in ASCII format.

GETDEVICECHECKSUM
Instructs the recipient to transmit a CRC16 checksum of its memory. This can be used to check the integrity of the programme memory after switching on.

RESET
Instructs the recipient to carry out a software reset. This resets the device to the switch-on state. The parameter is always 0.

Commands for the PLCS-21

The following table contains a list of the commands which the PLCS-21 supports in addition to the generally applicable commands. An explanation of the individual commands follows afterwards.

Command	Sent Frame		Received Frame	
	Command	Parameter	Command	Parameter
GETCPUTEMP	0x0001	0	0x0050	-50 … +200 °C
GETDEVTEMP	0x0002	0	0x0050	-50 … +200 °C
GETVOLMIN	0x0003	0	0x0053	0 … 4095
GETVOLMAX	0x0004	0	0x0053	0 … 4095
GETVOLSET	0x0005	0	0x0053	0 … 4095
GETVOLACT	0x0006	0	0x0053	0 … 4095
GETVOLPERSTEP	0x0007	0	0x0053	Refer to description
GETCURVAL	0x0008	0	0x0052	0 … 2^{31}-1 mA
GETLSTAT	0x0009	0	0x0054	0 … 2^{31}-1
GETDEVID	0x000A	0	0x0055	0 … 32
GETPULSEWIDTH	0x000B	0	0x0056	0 … 2^{31}-1 ns
GETPULSEWIDTHMIN	0x000C	0	0x0056	0 … 2^{31}-1 ns
GETPULSEWIDTHAX	0x000D	0	0x0056	0 … 2^{31}-1 ns
GETREPRATE	0x000E	0	0x0057	0 … 2^{31}-1 Hz
GETREPRATEMIN	0x000F	0	0x0057	0 … 2^{31}-1 Hz
GETREPRATEMAX	0x0010	0	0x0057	0 … 2^{31}-1 Hz
GETSHOTS	0x0011	0	0x0058	0 … 2^{31}-1
GETSHOTSMIN	0x0012	0	0x0058	0 … 2^{31}-1
GETSHOTSMAX	0x0013	0	0x0058	0 … 2^{31}-1
GETOVERCUR	0x0014	0	0x0052	0 … 4095
GETOVERCURMIN	0x0015	0	0x0052	0 … 4095
GETOVERCURMAX	0x0016	0	0x0052	0 … 4095
GETOVERCURVAL	0x0017	0	0x0052	0 … 2^{31}-1 (32 bit) mA
GETDEVTEMPOFF	0x001B	0	0x0050	16 bit short °C
GETDEVTEMPOFFMIN	0x001C	0	0x0050	16 bit short °C
GETDEVTEMPOFFMAX	0x001D	0	0x0050	16 bit short °C
GETUMIN	0x001E	0	0x0051	0 … 4095

Command	Sent Frame		Received Frame	
	Command	Parameter	Command	Parameter
GETERROR	0x001F	0	0x0059	0 … 2^{31}-1
GETDEVICENAME	0x0022	0	0x005C	Refer to description
SETVOL	0x0030	0 … 4095	0x0053	0 … 4095
SETLSTAT	0x0031	0 … 2^{31}-1	0x0054	0 … 2^{31}-1
SETREPRATE	0x0032	0 … 2^{31}-1	0x0057	0 … 2^{31}-1
SETPULSEWIDTH	0x0033	0 … 2^{31}-1	0x0056	0 … 2^{31}-1
SETSHOTS	0x0034	0 … 2^{31}-1	0x0058	0 … 2^{31}-1
SETOVERCUR	0x0035	0 … 4095	0x0052	0 … 4095
SETDEVTEMPOFF	0x0036	16 bit short	0x0050	16 bit short
SETUMIN	0x0038	0 … 4095	0x0053	0 … 4095
CLEARERROR	0x0039	0	0x005A	0
EXECCAL	0x003A	0	0x005B	Refer to text
RSTDEF	0x003C	0	0x0060	0

Description of the Individual Commands

GETCPUTEMP
Contains as return value the current temperature of the PLCS-21 in °C. The two low bytes of the answer parameter represent a "signed short".

GETDEVTEMP
Contains as return value the current temperature in °C of the pulser connected to the PLCS-21.
If no pulser is connected then the value is 0.

GETVOLMIN
Contains as return value the minimum voltage that can be set for the connected pulser. The range that can be set is divided into 4096 steps. If voltage is required in mV then a conversion factor can be requested using the command GETVOLPERSTEP.
If no pulser is connected then the value is 0.

GETVOLMAX
Contains as return value the maximum voltage that can be set for the connected pulser. The range that can be set is divided into 4096 steps. If voltage is required in mV then a conversion factor can be requested using the command GETVOLPERSTEP.
If no pulser is connected then the value is 0.

GETVOLSET
Contains as return value the current voltage specification for the connected pulser. The range that can be set is divided into 4096 steps. If voltage is required in mV then a conversion factor can be requested using the command GETVOLPERSTEP.
If no pulser is connected then the value is 0.

GETVOLACT
Contains as return value the measured value of the target voltage of the connected pulser. The range that can be set is divided into 4096 steps. If voltage is required in mV then a conversion factor can be requested using the command GETVOLPERSTEP.
If no pulser is connected then the value is 0.

GETVOLPERSTEP
Contains as return value a conversion factor with which the value range of 0…4095 can be converted to mV. The answer is a 64 bit double data word.
If no pulser is connected then the value is 0.

GETCURVAL
Contains as return value the presently set current in mA. This value is only valid if the PLCS-21 is in Current Mode and an up-to-date calibration is available. If no pulser is connected or if the PLCS is not in Current Mode, the value is 0.

GETLSTAT
Contains as return value the laser status register. There is a description of the individual bits further on in the document.

GETDEVID
Contains as return value the ID of the connected pulser.

GETPULSEWIDTH
Contains as return value the currently set pulse width in ns.

GETPULSEWIDTHMIN
Contains as return value the minimum settable pulse width in ns.

GETPULSEWIDTMAX
Contains as return value the maximum settable pulse width in ns.

GETREPRATE
Contains as return value the currently set repeat rate in Hz.

GETREPRATEMIN
Contains as return value the minimum settable repeat rate in Hz.

GETREPRATEMAX
Contains as return value the maximum settable repeat rate in Hz.

GETSHOTS
Contains as return value the currently set number of pulses to be emitted. This register is only used in the trigger modes 0 and 1.

GETSHOTSMIN
Contains as return value the minimum settable number of pulses to be emitted. This register is only used in the trigger modes 0 and 1.

GETSHOTSMAX
Contains as return value the maximum settable number of pulses to be emitted. This register is only used in the trigger modes 0 and 1.

GETOVERCUR
Contains as return value the presently set value for over-current detection. This is indicated in the range from 0…4095. If a value in mA is required, then this can be queried with GETOVERCURVAL. This register is only used if the PLCS-21 is not being used as frequency generator.

GETOVERCURMIN
Contains as return value the minimum settable value for over-current detection. This is indicated in the range from 0…4095.
This register is only used if the PLCS-21 is not being used as frequency generator.

GETOVERCURMAX
Contains as return value the maximum settable value for over-current detection. This is indicated in the range from 0…4095.
This register is only used if the PLCS-21 is not being used as frequency generator.

GETOVERCURVAL
Contains as return value the presently set value for over-current detection in mA.
This register is only used if the PLCS-21 is not being used as frequency generator.

GETDEVTEMPOFF
Contains as return value the presently set switch-off temperature in °C for the pulser which is connected. When this temperature is reached the pulser emissions are stopped and a temperature error is outputted.
This register is only used if the PLCS-21 is not being used as frequency generator.

GETDEVTEMPMIN
Contains as return value the minimum settable switch-off temperature in °C.
This register is only used if the PLCS-21 is not being used as frequency generator.

GETDEVTEMPMAX
Contains as return value the maximum settable switch-off temperature in °C.
This register is only used if the PLCS-21 is not being used as frequency generator.

GETUMIN
Contains as return value the presently set start voltage for the calibration. It is standardized to 0…4095 precisely the same way as the voltages in the VOLTAGE commands, and can be converted using the same factor. If a pulser calibration is carried out it will begin with this value.
This register is only used if the PLCS-21 is not being used as frequency generator.

GETERROR
Contains as return value the present content of the error register. There is a description of the individual bits further on in the document.

GETDEVICENAME
Instructs the recipient to send back a string which contains the name of the connected pulser. If 0 is sent as parameter, the answer contains the number of digits of the string, otherwise the respective position of the name is sent in ASCII format.

SETVOL
Changes the target specification of the set voltage. The parameter is standardized to 0…4095 and can be calculated from a value in mV using the conversion factor GETVOLPERSTEP. Only the range which is defined by GETVOLMIN and GETVOLMAX is permissible. Outside of this range the sender will receive a ILGLPARAM as answer. The answer parameter is the set target value.

SETLSTAT
Changes the laser status register to the passed parameter. The return value contains the changed status register.

SETREPRATE
Changes the repeat rate of the pulses to the passed value. Only the range which is defined by GETREPRATEMIN and GETREPRATEMAX is permissible. Outside of this range the sender will receive a ILGLPARAM as answer. The answer parameter is the set repeat rate.

SETPULSEWIDTH
Changes the width of the pulse to the passed value. Only the range which is defined by GETPULSEWIDTHMIN and GETPULSEWIDTHMAX is permissible. Outside of this range the sender will receive a ILGLPARAM as answer. The answer parameter is the set pulse width.

SETSHOTS
Changes the number of pulses to be emitted to the passed value. This register is only used in the trigger modes 0 and 1. Only the range which is defined by GETSHOTSMIN and GETSHOTSMAX is permissible. Outside of this range the sender will receive a ILGLPARAM as answer. The answer parameter is the number of pulses to be emitted.

SETOVERCUR
Changes the maximum permissible diode current to the passed value. Only the range which is defined by GETOVERCURMIN and GETOVERCURMAX is permissible. Outside of this range the sender will receive a ILGLPARAM as answer. The answer parameter is the set maximum current.

SETDEVTEMPOFF
Changes the switch-off temperature to the passed value. Only the range which is defined by GETDEVTEMPMIN and GETDEVTEMPMAX is permissible. Outside of this range the sender will receive a ILGLPARAM as answer. The answer parameter is the set maximum temperature.

SETUMIN
Changes the set minimum voltage for the calibration to the passed value. Only the range which is defined by GETVOLMIN and GETVOLMAX is permissible. Outside of this range the sender will receive a ILGLPARAM as answer. The answer parameter is the set minimum voltage.

CLEARERROR
Deletes the ERROR register and resets the PLCS to an error-free condition.

EXECCAL
Instructs the pulser to carry out a calibration. If the answer parameter is zero then a calibration is initiated by the PLCS-21. If it is unlike zero then it is currently not possible to carry out a calibration. This occurs when the PLCS is currently carrying out a calibration or if no pulser is connected.

RSTDEF
Instructs the PLCS-21 to reset all parameters to factory defaults. All calibration data will be lost. The default values are the minimum values for the PLCS-21 or the connected laser diode driver.

Description of the LSTAT Register

The following list contains a description of the individual LSTAT bits. These can be read with GETLSTAT and written with SETLSTAT. With SETLSTAT a complete 32 bit word must always be written. Thus, to change individual bits, first the register must be read out with GETLSTAT, then the desired bits changed and then with SETLSTAT passed again to the PLCS.

Bit	Name	Read/Write	Meaning
0	L_ON	Read/write	Switch on/off the pulse output
1	MODE	Read	Operating mode: 0 = Normal 1 = Frequency Generator
2-5	TRG_MODE	Read/write	Refer to trigger modes
6	ENABLE_HELPPULSE	Read/write	Reserved
7	EMABLE_FEEDBACK_MON	Read/write	Reserved
8	VOLTAGEMODE	Read/write	Switches between Voltage Mode and Current Mode
9	UNCAL	Read/write	Indicates whether calibration data is available
10	CALIBRATING	Read	Indicates that a calibration is currently being carried out
11	Reserved	Read	Reserved
12	BUSY	Read	Indicates that the PLCS is currently not accepting commands
13	INIT_COMPLETE	Read	Indicates successful initialization
14	DEVICE_CHANGED	Read	Indicates that another device type has been connected since the last start
15-31	Reserved	Read	Reserved

The following table contains the appropriate LSTAT bits for the trigger modes described in chapter "Trigger Modes".

Mode	LSTAT Bits				Description
	Bit 5	Bit 4	Bit 3	Bit 2	
0	0	0	0	0	External trigger, falling edge. Number of pulses can be set
1	0	0	0	1	External trigger, rising edge. Number of pulses can be set
2	0	0	1	0	Internal trigger, ongoing pulses
3	0	0	1	1	Internal trigger, ongoing pulses
4	0	1	0	0	External trigger, pulse output during a LOW level on the trigger input
5	0	1	0	1	External trigger, pulse output during a HIGH level on the trigger input

Description of the ERROR Register

The following list contains a description of the individual bits of the ERROR register. A "1" as a bit leads to a deactivation of the pulser output. Bit 5 and 10 are excluded of this directive. They must be acknowledged with CLRERROR before pulse output can take place again. If the bits 9, 12 or 15 occur then the voltage supply must be briefly disconnected to delete the bits.

Bit	Name	Read/Write	Meaning
0	IMAX_OVERSTEPPED	Read	The PLCS measured a pulse current larger than the set value and has switched off
1	VOLTAGE_FAIL	Read	Reserved
2	Reserved	Read	Reserved
3	CPUTEMP_OVERSTEPPED	Read	The PLCS has exceeded its permitted operating temperature of 80 °C
4	Reserved	Read	Reserved
5	DEVICETEMP_WARN	Read	The temperature of the connected pulser is approaching the switch-off temperature
6	DEVICETEMP_OVERSTEPPED	Read	The switch-off temperature of the connected pulser has been reached
7	DEVICETEMP_HYSTERESIS	Read	The connected pulser is currently in the cooling phase
8	DEVICETEMP_SENSORFAILED	Read	The temperature sensor in the pulser is defective
9	DEVICE_FAILED	Read	The connected pulser is defective
10	NODEVICE	Read	No connected pulser was detected
11	CALERROR	Read	An error occurred during calibration
12	TBL_FAIL	Read	No data was found for the connected pulser, a software update is necessary
13	Reserved	Read	Reserved
14	Reserved	Read	Reserved
15	U_15V_FAIL	Read	The supply voltage is too low
16	INTERNAL_ERROR	Read	An internal error occurred. Please contact support
17	FAULTY_ID	Read	The connected pulser has an invalid ID
18-31	Reserved	Read	Reserved

If a critical error occurs pulser emissions stop automatically. All error situations must be acknowledged or reset with CLRERROR. Otherwise the PLCS cannot restart pulse output.

Example Implementation in MS Visual Basic

The following is a possible implementation of the protocol for uni-directional communications in MS Visual Basic. No guarantee of functionality is assumed.

```
Public Class Protocol
    Public Const PING As UShort = &HFE01
    Public Const IDENT As UShort = &HFE02
    Public Const GETHARDVER As UShort = &HFE06
    Public Const GETSOFTVER As UShort = &HFE07
    Public Const GETSERIAL As UShort = &HFE08
    Public Const GETIDSTRING As UShort = &HFE09
    Public Const GETDEVICECHECKSUM As UShort = &HFE0B
    Public Const RESET As UShort = &HFE0E
    Public Const ACK As UShort = &HFF01
    Public Const IDACK As UShort = &HFF02
    Public Const VERSIONACK As UShort = &HFF03
    Public Const HARDVERACK As UShort = &HFF06
    Public Const SOFTVERACK As UShort = &HFF07
    Public Const SERIALACK As UShort = &HFF08
    Public Const IDSTRINGACK As UShort = &HFF09
    Public Const CHECKSUMACK As UShort = &HFF0A
    Public Const RESETACK As UShort = &HFF0B
    Public Const RXERROR As UShort = &HFF10
    Public Const REPEAT As UShort = &HFF11
    Public Const ILGLPARAM As UShort = &HFF12
    Public Const UNCOM As UShort = &HFF13

    Private RecParameter As UInt64 = 0
    Private RecAnswer As UInt64 = 0
    Private Comport As String = ""
    Private PortOpen As Boolean = False
    Private Serial As IO.Ports.SerialPort = Nothing
    Private IamBusy As Boolean = False

    Public Function GetAnswer() As UShort
        Return RecAnswer
    End Function

    Public Function GetParameter() As UInt64
        Return RecParameter
    End Function

    Property Status() As Integer
        Get
            Return PortOpen
        End Get
        Set(ByVal Value As Integer)

        End Set
    End Property

    Property Busy() As Integer
        Get
            Return IamBusy
        End Get
        Set(ByVal Value As Integer)
        End Set
    End Property

    Public Function Enable(ByVal port As String) As Boolean
```

```
        If (PortOpen) Then
            Return True
        End If

        Try

            If (Not (port = "")) Then
                Comport = port
            End If
            Serial = New IO.Ports.SerialPort(Comport, 115200,
IO.Ports.Parity.Even, 8, IO.Ports.StopBits.One)
            Serial.Open()
            PortOpen = True

            SendReceive(Me.PING, 0, Me.ACK)
            SendReceive(Me.PING, 0, Me.ACK)

        Catch ex As Exception

            PortOpen = False
            Return False

        End Try

        Return True
    End Function

    Public Function Disable() As Boolean
        If (PortOpen) Then
            Try
                Serial.Close()

            Catch ex As Exception

            End Try

            PortOpen = False
            Serial = Nothing

            Return True

        End If

        Return False
    End Function

    Public Function SendReceive(ByVal command As UShort, ByVal param As
UInt64, ByVal expectet_answer As UShort) As Boolean

        Dim Timeout As UInt32 = 10000
        Dim buffer(12) As Byte

        If (Not PortOpen) Then
            Return False
        End If
```

```
        If (IamBusy) Then
            Do
                Application.DoEvents()
            Loop While IamBusy = True
        End If

        IamBusy = True

        For i As UInteger = 0 To 4
            Timeout = 10000

            Serial.DiscardInBuffer()

            Send(command, param)

            Do
                Timeout -= 1
                Application.DoEvents()
            Loop Until ((Serial.BytesToRead() >= 12) Or (Timeout = 0))

            If (Timeout > 0) Then

                If (Serial.BytesToRead() >= 12) Then

                    If (Receive(buffer)) Then
                        RecAnswer = buffer(0)
                        RecAnswer += Convert.ToUInt16(buffer(1)) << 8
                        RecParameter = buffer(2)
                        RecParameter += Convert.ToUInt64(buffer(3)) << 8
                        RecParameter += Convert.ToUInt64(buffer(4)) <<
16
                        RecParameter += Convert.ToUInt64(buffer(5)) <<
24
                        RecParameter += Convert.ToUInt64(buffer(6)) <<
32
                        RecParameter += Convert.ToUInt64(buffer(7)) <<
40
                        RecParameter += Convert.ToUInt64(buffer(8)) <<
48
                        RecParameter += Convert.ToUInt64(buffer(9)) <<
56
                        IamBusy = False

                        Return (RecAnswer = expectet_answer)
                    End If

                End If

            End If

        Next
        IamBusy = False

        Return False

    End Function

    Private Function Send(ByVal command As UShort, ByVal param As
UInt64) As Boolean

        Dim buffer(12) As Byte
```

```
        buffer(0) = command And &HFF
        buffer(1) = (command >> 8) And &HFF
        buffer(2) = param And &HFF
        buffer(3) = (param >> 8) And &HFF
        buffer(4) = (param >> 16) And &HFF
        buffer(5) = (param >> 24) And &HFF
        buffer(6) = (param >> 32) And &HFF
        buffer(7) = (param >> 40) And &HFF
        buffer(8) = (param >> 48) And &HFF
        buffer(9) = (param >> 56) And &HFF
        buffer(10) = 0

        buffer(11) = CheckByte(buffer)

        WriteByte(buffer)

    End Function

    Private Function Receive(ByVal buffer() As Byte) As Boolean

        For i As UInteger = 0 To 11 Step 1
            buffer(i) = ReadByte()
        Next

        If (buffer(11) = CheckByte(buffer)) Then
            Return True
        End If

        Return False

    End Function

    Private Function CheckByte(ByVal buffer() As Byte) As Byte
        Dim returnvalue As Byte = 0

        For i As UInteger = 0 To 10 Step 1
            returnvalue = returnvalue Xor buffer(i)
        Next

        Return returnvalue
    End Function

    Private Sub WriteByte(ByVal zeichen() As Byte)
        Serial.Write(zeichen, 0, 12)
    End Sub

    Private Function ReadByte() As Byte
        Return Serial.ReadByte()
    End Function

End Class
```

Using this example code, a connection can be set up using the following lines of code:

```
Dim MyProto As Protocol = New Protocol()

MyProto.Enable(„Com3“)

MyProto.SendReceive(Protocol.PING, 0, Protocol.ACK)
```

Electrical Characteristics

Parameter	Symbol	Condition	Min.	Typ.	Max.	Unit
Supply current		U_S=15V, no cable/device connected to PLCS-21		175		mA
Load resistance (con. 3)	R_L		10	50		Ohm
Output voltage (con. 3)	U_L	R_L=50 Ohm	4.85	5	5.15	V
Input resistance (con. 5)	$R_{T,50}$		48	50	52	Ohm
Input resistance (con. 4)	$R_{T,500}$		485		515	Ohm
Low Level input Voltage (con. 5)	$U_{T,50}$	U_S=15V			0.5	V
High Level input Voltage (con. 5)	$U_{T,50}$	U_S=15V	0.6	2.3	3.4	V
Low Level input Voltage (con. 4)	$U_{T,500}$	U_S=15V			0.5	V
High Level input Voltage (con. 4)	$U_{T,500}$	U_S=15V	0.9	3.5	4.7	V

Absolute Maximum Ratings

Parameter (see figures)	Symbol	
Ambient operating temperature		0°C to +55°C
Supply voltage	U_S	-0.3V to +15.5V
Trigger voltage on connector 5	$U_{T,50}$	-6V to +6V
Trigger voltage on connector 4	$U_{T,500}$	-6V to +6V
Load current on connector 3	I_L	170mA